Amendment to the Specification:

Please delete the paragraph beginning at page 1 line 4, which starts with "A method for data flow control..." and ends at page 1 line 5 with "... in a packet data transmission system."

Please add the following new paragraphs after the title:

CROSS-REFERENCE TO RELATED APPLICATIONS

This is a National Stage Application of International Patent Application No. PCT/PL 03/00083, with an international filing date of August 25, 2003, which is based on Polish Patent Application No. P-355743, filed August 28, 2002.

BACKGROUND OF THE INVENTION

1. Field of the Invention

Please add the following new subheading on page 1, after line 8, (i.e., after the text ending with "... in formats used in digital television"):

2. Brief Description of the Background of the Invention Including Prior Art

Please amend the paragraph starting at page 1, line 14, as follows:

There are also known methods for filtering and manipulation of packets, ~~flowing~~ transmitted through a given network. For example, commonly used firewalls software (e.g. ipchains from Linux) enable to set packet processing rules, for example taking into account their source, destination or type, depending on whether packets are incoming or outgoing ones. However, the rules available in present software allow mainly filtering and directing packets to appropriate outputs, but do not allow conversion of data that the packets carry.

Please add the following new paragraph on page 2, before line 1, (i.e., before the paragraph beginning with "It is the object of the present invention to create a method for control..."):

SUMMARY OF THE INVENTION

1. Purposes of the Invention

It is an object of this invention to provide a method for data flow control in a packet data transmission system and a device for data flow control in a packet data transmission system that

allow filtering, directing and conversion of data packet transmitted through the transmission system more efficient than know systems.

This and other objects and advantages of the present invention will become evident from the description which follows.

2. Brief Description of the Invention

Please amend the paragraphs starting at page 2, line 1, as follows:

It is the object of the present invention to create a method for control of data flow in a packet data transmission system and a device for control of data flow in a system, where the data flow control components and packet manipulation components are nodes. The nodes may be in a form of input nodes (supplying data in a required format), multiplexing nodes (joining several transmission unit streams into one stream) or output nodes (performing transmission unit conversion to a given output format). Besides their basic functions, such as conversion and multiplexing, each node can be assigned rules characterizing data management commands, which are to be applied to transmission units, ~~flowing through~~ processed in the node.

The rules define transmission units to be modified, by specifying a label, a type or a size of a packet included in a given transmission unit, and also define methods of execution of such modifications by providing an appropriate command. The rules may also determine a need for a conversion of specific packets in specific transmission units to a different format. The packets entering the system are converted into transmission units, which are created by encapsulating the input packets by adding a label, a field determining the packet type and a field determining the packet size. The label is used to mark the input, which the transmission unit (and so the packet contained therein) was ~~retrieved~~ read from. Unit labels can be modified in every system node.

The method for control of data flow in a packet data transmission system according to the invention is based on the fact that data is supplied to the input nodes of the processing system comprising a network of nodes in the form of input nodes, output nodes and intermediate nodes such as data processing nodes or multiplexers, where the nodes are connected in a user-defined structure. The packet data supplied to the input nodes of the system is converted into transmission units, and each of the nodes is assigned input and output rules, as well as general rules.

Please amend the paragraphs starting at page 3, line 2, as follows:

Whenever a ~~packet~~ transmission unit is available at the node input, it is checked whether the general rules apply to this unit and in case of a positive result of this check, the commands, determined by these rules, are executed and then it is checked whether the input rules apply to this unit. If yes, the commands, determined by these rules, are executed and then the node function is performed, followed by a check whether the output rules apply to this unit. In the output nodes, the packets are extracted from transmission units, ~~which are~~ the transmission units having been created by adding a label field, a type field and/or a size field to the packet. In case when a rule is a conversion rule, it is checked whether a given conversion algorithm requires additional rules being present. If it does and the additional rules are not present, the packet is rejected. The packets, to which the given rule applies, are defined by their label, type, size or similar parameters.

The device for control of data flow in a packet data transmission system according to the present invention, which uses the above described method for control of data flow, comprises at least two input nodes, processing supplied data, forming transmission units and provisionally converting input packet ~~signal~~ data from a given format to another, which is performed on the basis of output rules set for input nodes. The outputs of input nodes are connected to other system nodes, including at least one multiplexer and/or at least one data processing node and/or at least one output node. The nodes process the ~~flowing~~ transmission units based on the general, input and output rules set for nodes according to the data flow control method described above. The outputs of the multiplexers and data processing nodes are connected to other multiplexers or data processing nodes or output nodes. The output nodes are the final nodes in the system and in these nodes the transmission units are converted to the packets they carry.

Please amend the paragraph starting at page 4, line 2, as follows:

The above-described device and method for control of data flow in a packet data transmission system create a possibility of a convenient control of data flow through the system. The essential features are the possibility to encapsulate packet data by adding a label, a type and a packet size fields in the system input nodes and the possibility of removing this information in the output nodes as well as extended possibilities of packet processing in the system nodes allowing packet processing at every input and output of every node, manipulating the packets according to the general rules or local rules assigned to particular inputs and outputs of system nodes,

distinguishing packets based on their label, type and/or size and finally filtering packets and their conversion to a given format in every system node.

Please add the following new paragraph on page 4, after line 22, (i.e., after the paragraph ending with "...a predetermined structure and identification fields, for example a PID field."):

The novel features which are considered as characteristic for the invention are set forth in the appended claims. The invention itself, however, both as to its construction and its method of operation, together with additional objects and advantages thereof, will be best understood from the following description of specific embodiments when read in connection with the accompanying drawings.

Please delete the paragraph beginning at page 4 line 23, which starts with "The object of the present invention is shown as an exemplary embodiment..." and ends at page 5 line 9 with "... algorithm of output nodes operation."

Please add the following new paragraphs on page 5, before line 10, (i.e., before the paragraph beginning with "As shown in Fig. 1 of the drawing..."):

BRIEF DESCRIPTION OF THE DRAWINGS

In the accompanying drawings one of the possible embodiments of the present invention is shown, where:

Fig. 1 shows a general block structure of a data flow control device;

Fig. 2 shows an exemplary block diagram of a data flow control device;

Fig. 3 shows an exemplary node of a single multiplexer and a list of rules assigned to the exemplary node;

Fig. 4 shows a block diagram of an algorithm of transmission unit creation in an input node;

Fig. 5 shows a structure of a transmission unit, schematic diagram of an extended embodiment of a signal controller

Fig. 7 shows a block diagram of a single node operation algorithm;

Fig. 8 shows a block diagram of a rules management algorithm;

Fig. 9 shows a block diagram of an algorithm of transmission unit data conversion to a required output format; and

Fig. 10 shows a block diagram of an algorithm of output nodes operation.

DESCRIPTION OF INVENTION AND PREFERRED EMBODIMENT

Please amend the paragraph starting at page 5, line 10, as follows:

~~As shown in fig. 1 of the drawing,~~ Fig. 1 illustrates an exemplary structure of a device for control of data flow in a packet data transmission system. This device is a network of input nodes **IN1 - INn**, output nodes **OUT1 - OUTn** and intermediate nodes that are data processing nodes **PROC1 - PROCn** or multiplexers **MUX1 - MUXo**. These nodes are connected in a user-defined structure. Signals in form of data packets are supplied to input nodes **IN1 -INn** and from data packets ~~retrieved~~ read from these nodes **IN1 - INn** transmission units are created. Every node **IN1 - INn, PROC1 - PROCn, MUX1 - MUXo** and **OUT1 - OUTn** has input rules **RIN1 - RINn, RPROC1 - RPROCn, RMUX1- RMUXo** and output rules **ROUT1 - ROUTn** assigned to it beforehand. The system also has general rules defined. Moreover, in the input nodes **IN1 - INn** the transmission units are assigned labels, which identify the input node **IN1 - INn**, from which ~~a given unit originates~~ the units originate.

Please amend the paragraph starting at page 5, line 22, as follows:

Fig. 2 provides further details for network structure and presents an exemplary block diagram of a device for control of data flow in a packet data transmission system. According to the figure, the device comprises six input nodes **IN1 - IN6**, processing ~~retrieved~~ read data, forming from them transmission units and provisionally converting ~~signals of~~ input packets from a one format to another. The outputs of the first three input nodes **IN1 - IN3** are connected to inputs of the first multiplexer **MUX1**, the outputs of the further two input nodes **IN4 - IN5** are connected to the inputs of the second multiplexer **MUX2** and the output of the last input node **IN6** is connected to one of the inputs of the next multiplexer **MUX3**, to the second input of which the output ~~signal~~ data from the second multiplexer **MUX2** is transmitted. The output of the first multiplexer **MUX1** is connected to the input of the processing node **PROC1**, which performs programmed actions on the data it receives and its output ~~signal~~ data is passed to one of the inputs of yet another multiplexer **MUX5**, to the second input of which the output ~~signal~~ data from multiplexer **MUX3** is transmitted. The multiplexer **MUX5** output ~~signals~~ data are passed to the three output nodes **OUT1 - OUT3**, which according to set rules block or pass packets having

predetermined labels, for example in order to adapt them to a given service that the output node prepares data for.

Please amend the paragraphs starting at page 6, line 18, as follows:

The data entering the transmission system is processed in the input nodes **IN1 - IN6**, in which transmission units are created from the input data. In the nodes **IN1 - IN6** a provisional conversion of packets may be carried out from one format to another. For example, the node **IN3** has MPEG format set as the output format and the inputs are data files, so in this node a conversion from data files to MPEG format is performed. Similar conversion is carried out in the node **IN5**. In the input nodes **IN1 - IN6** the packets are assigned labels, which may later be useful to determine from which of the nodes **IN1 - IN6** the data comes. Accordingly, **IN1** assigns **L1** to all its packets; **IN2** assigns **L2** and so on, to the **L6** being assigned to all **IN6** packets. It is also possible to define other operations on packets, for example changing identification data or omitting all or certain packets, which allows disabling a given input. The transmission units from input nodes **IN1**, **IN2** and **IN3** are combined into one stream by the multiplexer **MUX1**, for which rules, such as for example passing or blocking units of a given label, have been previously set. The output format of units of **L1** label from node **MUX1** is MPEG, so data originating from mode **IN1** (PS – Private Section – a data format related to MPEG format) will be converted to MPEG packets. The output data of the multiplexer **MUX1** are sent to the processing node **PROC1**, where programmed actions are performed on ~~retrieved~~ read data.

The node **PROC1** has also its rules assigned. Its output ~~signal~~ data is passed to the multiplexer **MUX5**. The units labeled **L1** and **L2** that go via **MUX5** are converted to the DARC (Data Radio Channel) format, while the remaining units (**L3**, **L4**, **L5** and **L6**) are converted to the MPEG format if such conversion is required. Similar conversion rules are set for packets having certain data, type or size. The multiplexer **MUX5** output data is passed to three different output nodes **OUT1**, **OUT2** and **OUT3**. These nodes can also have predetermined sets of data processing rules such as for example blocking or passing packets having certain labels in order to adapt them to a given service, for which the output node prepares data.

Please amend the paragraphs starting at page 8, line 11, as follows:

The algorithm of processing of system input data and the data ~~flowing~~ transmitted inside the system will now be described. In the input nodes the input data is encapsulated to form a

transmission unit, which is presented in ~~fig.~~ Fig. 4 and the structure of the transmission unit is shown in ~~fig.~~ Fig. 5. The content of the transmission unit can be converted to a given format, which can take place in any node, according to the algorithm shown in ~~fig.~~ Fig. 6. For the ease of data flow control, uniform packets, in the form of transmission units, are created in which packets of any format can be transmitted.

Please amend the paragraphs starting at page 14, line 6, and ending at page 16, line 18, as follows:

A general node operation algorithm is shown in ~~fig. 6 of the drawing~~ Fig. 6. The first step 301, for a node, is to read the input data ~~301~~, i.e. the input transmission unit. Then a check is made in step 302 whether general rules are defined ~~302~~. The general rules are checked for input data. Therefore, if the general rules cause skipping of certain units, other rules or functions of a given node will not process these units. Next the rules management algorithm is executed in step 303, which receives a list of general rules of a given node as its input data and executes them for the transmission unit currently processed in the node. The algorithm is shown in ~~fig. 7 of the drawing~~ Fig. 7. Then in step 304, a check is made whether there are any input rules, for the input, which the read transmission unit originates from ~~304~~. If yes, the rules management algorithm is executed in step 305, by passing to it a list of input rules that apply to a given transmission unit. Next, the node functions are executed in step 306, such as data processing or multiplexing. Then in step 307 a check is made whether there are any output rules, for the output, which the transmission unit will be directed to ~~307~~. If yes, the rules algorithm is executed in step 308, by passing to it a list of rules that apply to a given transmission unit. The last step 309, involves sending the transmission unit. It is obvious, as it has already been mentioned, that for the output nodes the steps 302 – 305 are not performed and for the input nodes the steps 307 – 308 are not performed, since there would be no transmission units for them.

Fig. 7 ~~of the drawing~~ shows the rules management algorithm. It starts from the first rule assigned for a given transmission unit in step 401. The rule is read in step 402 and commands defined by the rule are executed. If there is a conversion command defined in the rule, the algorithm shown in ~~fig.~~ Fig. 9 is performed, otherwise the algorithm shown in ~~fig.~~ Fig. 8 is performed. Next in step 404, a check is made whether as an outcome of applying the rule the unit has been split into a number of smaller units ~~404~~ and if the currently executed rule does not concern conversion. If both conditions are fulfilled, the rule is performed for the subsequent units

in step 405. Then the procedure checks whether there are any further rules for the current unit (or the units created after current unit splitting) in step 406. If there are, they are performed in step 407. If there are no more rules, the rules management for a given unit is finished in step 408.

Fig. 8 presents an algorithm of execution of commands defined by rules, except the rules with conversion command, for which the algorithm is presented in ~~fig.~~ Fig. 9. First a rule is read in step 421, followed by reading transmission unit information in step 422, such as the data type or identifying data of the data packet contained in the transmission unit. Then in step 423 a check is made whether the unit type allows execution of the action defined by the rule ~~423~~ (for example, when the rule applies to change of identification data, a check is made whether there are specific identification data in the unit). If it does, the action defined by the rule is performed in step 424 (for example, a change of packet identification data or removal of a packet). This completes the execution of commands for a given unit.

Fig. 9 presents a method of conversion of packets encapsulated in transmission units. The procedure described below is invoked by a *Convert* command. In its first step 451 the procedure reads the format of the ~~input~~ data packet ~~451~~ contained in the transmission unit. Then in step 452, a check is made whether there is an algorithm that converts packets of an input format to an output format defined for the node ~~452~~. If not, the ~~packet~~ the transmission unit is rejected in step 455. Otherwise, the procedure checks whether a given algorithm requires the presence of any further rules, and if it does then in step 453 it checks whether these rules are defined ~~453~~ (for example, there may be a rule required that assigns a specific value to a given field of a packet header). If the check fails, the unit is rejected. Otherwise, the conversion algorithm is executed in step 454. The actions of the conversion algorithm depend on the conversion type and are outside of the scope of this description. The conversion may involve a change of data that a transmission unit includes, an increase of unit size by adding zero bits or a split of the unit into smaller units. In case of a split, the unit header is assigned to every new packet according to the algorithm shown in ~~fig. of the drawing~~ Fig. 4.

The output nodes are responsible for the generation of a stream in an appropriate format. This is done in step 306 of the algorithm shown in ~~fig.~~ Fig. 6, after the commands of the general rules and input rules for a given input have been performed.

The execution of functions of the output nodes is performed according to the procedure shown in ~~fig. of the drawing~~ Fig. 10. A transmission unit is read in step 501. Then, a packet is extracted from the unit and label, type and size fields are removed in step 502. The next step 503

involves performing further actions on the packet, defined for a given output node (for example, conversion of the packet to a different format). After those actions are finished, the packet is ready for transmitting in step 504 and the procedure returns to step 309 of ~~fig.~~ Fig. 6, where the packet is transmitted.

Please add the following new paragraphs on page 16, after line 18, (i.e., after the paragraph ending with "... where the packet is transmitted."):

It will be understood that each of the elements described above, or two or more together, may also find a useful application in other types of methods for data flow control and devices for data flow control from the types described above.

While the invention has been illustrated and described as embodied in the context of a method for data flow control in a packet data transmission system and a device for data flow control in a packet data transmission system, it is not intended to be limited to the details shown, since various modifications may be made without departing in any way from the spirit of the present invention.

Without further analysis, the foregoing will so fully reveal the gist of the present invention that others can, by applying current knowledge, readily adapt it for various applications without omitting features that, from the standpoint of prior art, fairly constitute essential characteristics of the generic or specific aspects of this invention.

What is claimed as new and desired to be protected by Letters Patent is set forth in the appended claims.